



リモートセンシング工学 (第1回：ガイダンス)

小槻 峻司

(shunji.kotsuki@chiba-u.jp)

千葉大学・環境リモートセンシング研究センター
准教授

2020年度 千葉大学・工学部・情報工学コース 小槻担当①

Who am I ? :: Shunji Kotsuki (小槻 峻司)

高知

(1986)

京都大学・工学部
(2005)



理研・計算科学
(2014)

Postdoc

Researcher

千葉大・CEReS
(2019)

Associate Prof.



文科省・卓越研究員
(2017-2019)



JST・さきがけ研究員 (2019-)
理研計算・客員研究員 (2019-)



専門: 世界の天気予報

http://www.eorc.jaxa.jp/theme/NEXRA/index_j.htm



- JAXA, 東大, 理研と共同開発。10年後の天気予報を創る。
- スパコン「富岳」も利用。
- この分野の研究成果は、社会生活に直結！

授業の狙い

• 授業の目標

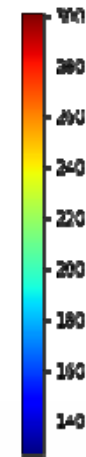
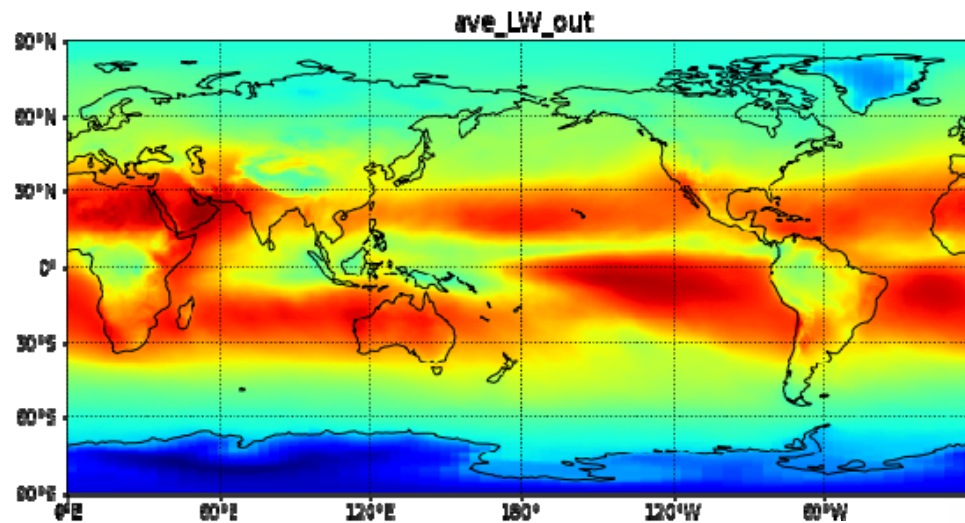
- 天気予報の仕組み（データ同化）や、その中で使われているリモートセンシング（人工衛星）データを理解する
- 天気の特徴について、教科書に出てくる図を自分で作成しながら体得する。
- その過程で、"使える"データ処理技術を習得する

• 実施する事

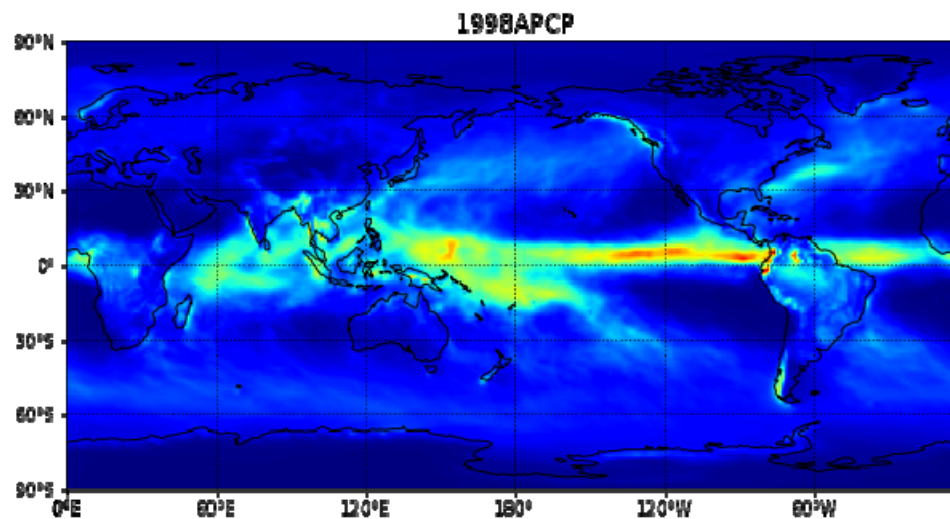
- 各回（今回を除く）、講義前に教科書を事前学習
 - 30分程度の学習量
 - 教科書は 小倉義光「一般気象学」（名著！）
- 該当箇所のミソを小槻が講義で解説
 - 動画30分程度
- 課題に取り組む
 - 30分程度の演習量。
- ソースコードとレポートを提出 (moodle)



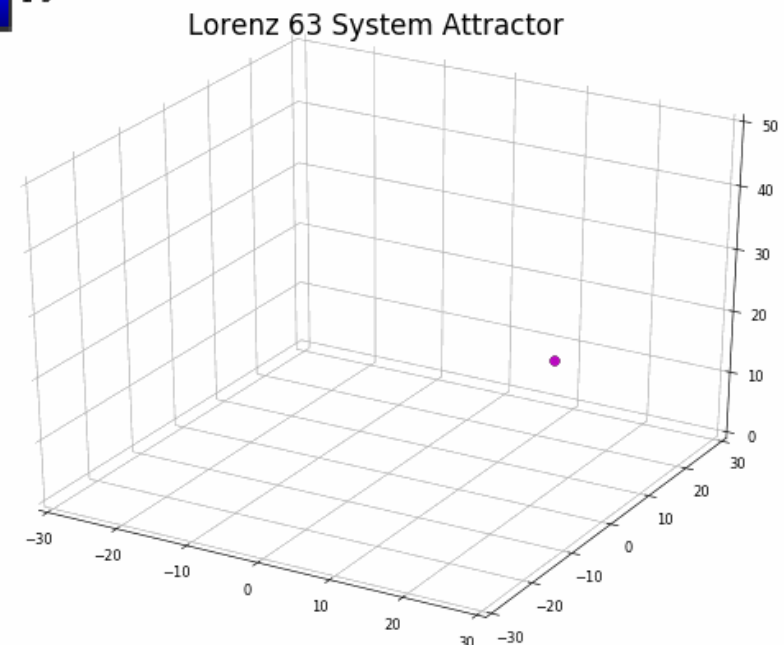
こんなことが出来るようになる！



**例 1：地球大気の赤外放射
≡どこに雲が多いか？**



例 2：世界の降水量分布 (1998年)



**例 3：ローレンツのバタフライ
アトラクタとカオス**

プログラミング課題・レポートについて

• プログラミング課題

- google colaboratoryにて、pythonで実施
 - 無料。後で使い方を解説
 - 今の時代、pythonの学習は必須です！
- pythonの基礎が全くない場合は、小槻研究室で整備中のpythonマニュアルを使って補助学習できます。
 - moodle から取得可能です
- どうしてもプログラミングが苦手であれば、各回、プログラミング演習の代わりにA4で1枚のレポートでもOKとします。ただし、最大で90%までしか配点しません。

• プログラミングとレポートについて

- 引用の無いcopy&pasteや、他の学生と全く同じソースコードが提出された場合、該当回は0点にします。
- “身に着ける”事が目的です。自分の手を動かそう！

各回の予定・その1（小槻担当）

- **第1回：ガイダンス & 天気予報とリモセン**

- 演習：numpy, matplotlibの基礎の習得
- 雑談：なんで学習するのか？ 心のしくみより

- **第2回：大気の鉛直構造**

- 演習：大気温度の鉛直プロファイルの解析・分析
- 雑談：ハーバーボッシュ法とその裏側の歴史

- **第3回：大気の熱力学と降水過程**

- 演習：降水量・蒸発散量の平衡を可視化
- 雑談：行動経済学とプロスペクト理論

- **第4回：大気の放射**

- 演習：地球の短波・長波放射とその平衡を可視化
- 雑談：生物と無生物の間、ウィルスと感染の人類史

各回の予定・その2（小槻担当）

- **第5回：大気の運動**

- 演習：地球3循環（ハドレー, フェレル, 極）を描く
- 雑談：日本の地形。中央構造線とその付加体

- **第6回：メソスケールの気象**

- 演習：衛星の視た平成30年7月豪雨を解析
- 雑談：「科学」ってなんだろう？

- **第7回：気候の変動**

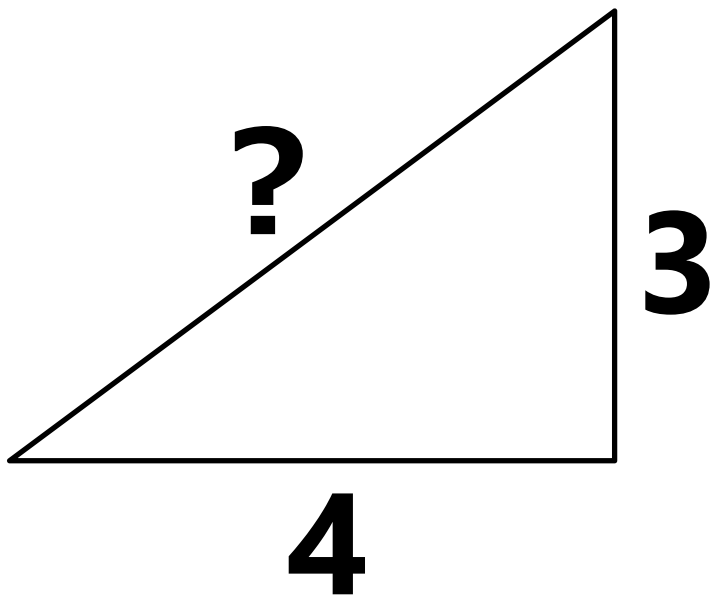
- 演習：地球大気のEOF解析とエルニーニョ現象
- 雑談：研究室を選ぶモノサシ、小槻研究室の紹介

- **第8回：数値天気予報の仕組みとデータ同化**

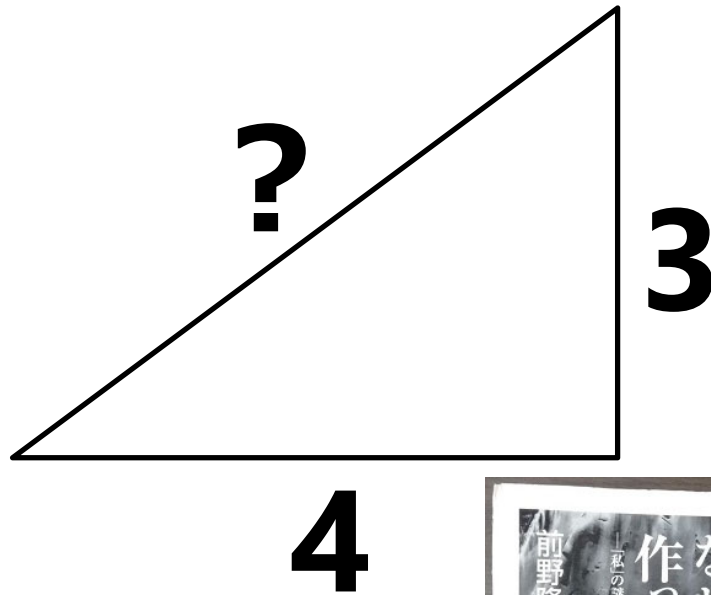
- 演習：ローレンツ・バタフライとカオスを描く
- 雑談：人生ってデータ同化

今日の雑談：
なんで学習するのか？
（“こころ”の仕組みより）

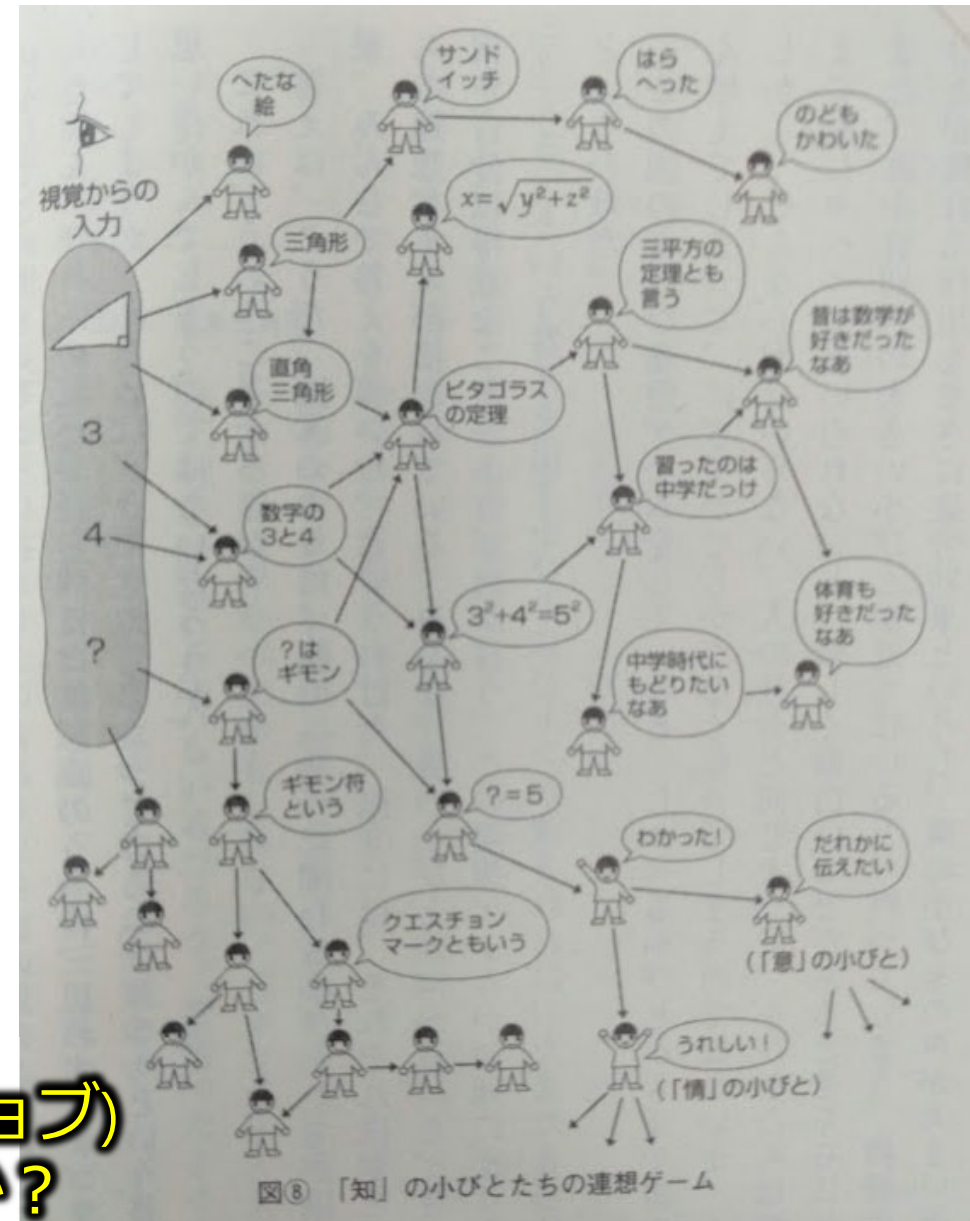
これなんですか？



脳の仕組み（個人的に腑に落ちた理解）



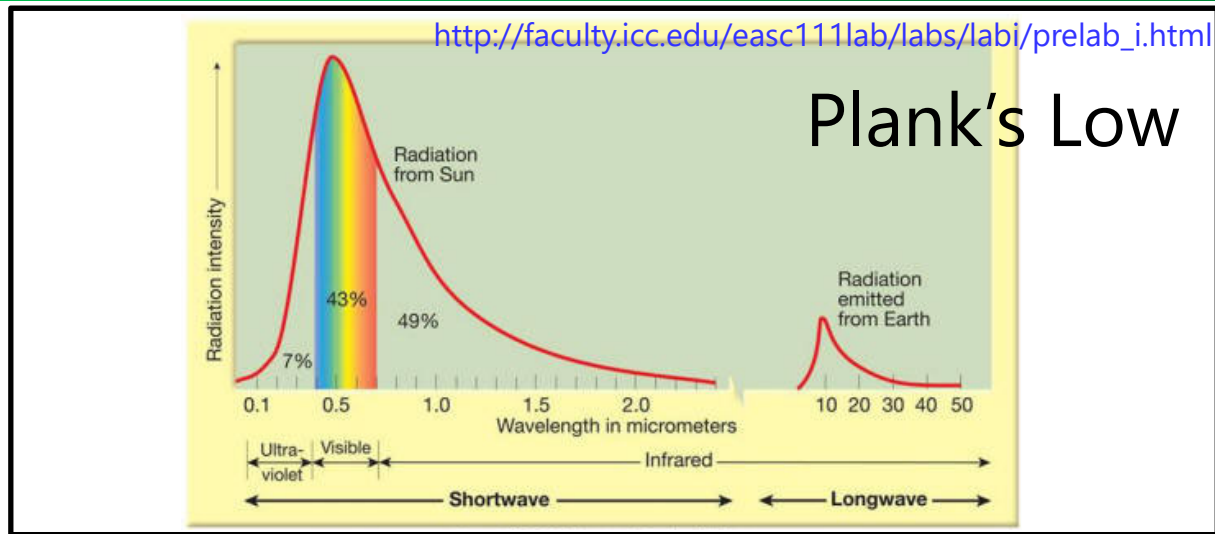
脳は小人(バックグラウンドジョブ)の連想ゲームなのではないか？



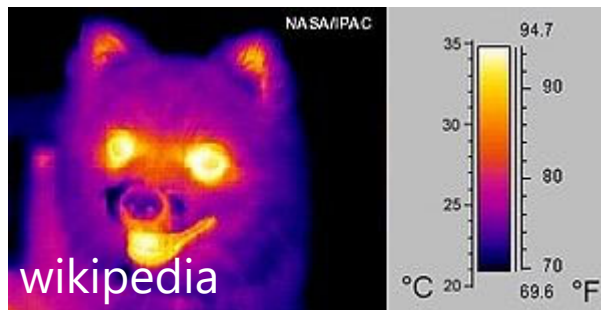
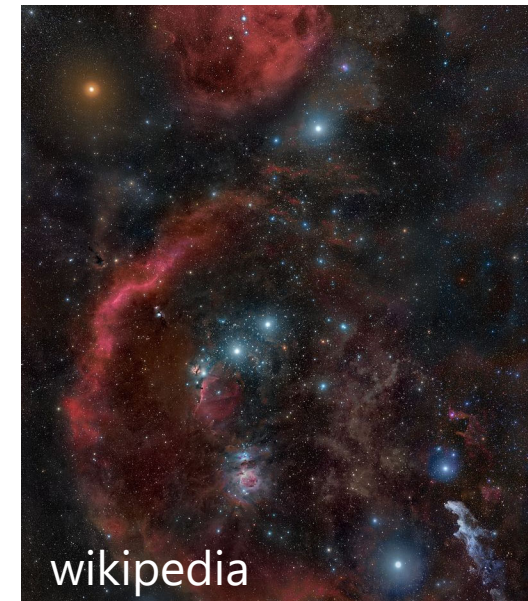
バックグラウンドジョブが有効に使えると？

- 運が良い人は、本当に、運が良いのか？
- 虫の知らせは、本当に、虫の知らせなのか？
- 意識に昇らないものの、バックグラウンドジョブが情報を処理し続けていて、それがあ
る時、突然意識に昇るのではないか？
- では、バックグラウンドジョブはどうやったら鍛えられるの？
- 小槻's answer : “結びつけ”の経験

If you understand radiation, then we can understand

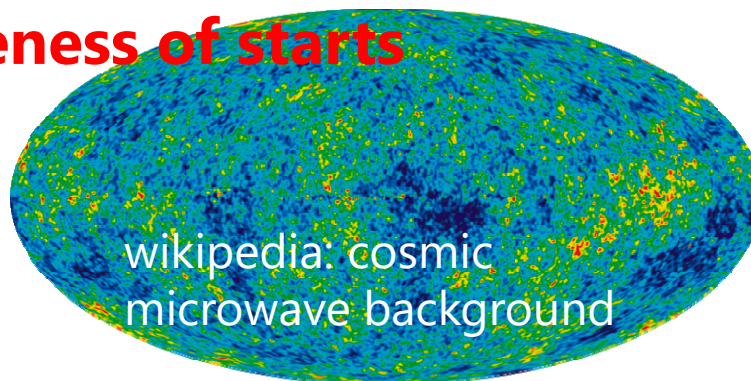


red Betelgeuse & white Sirius

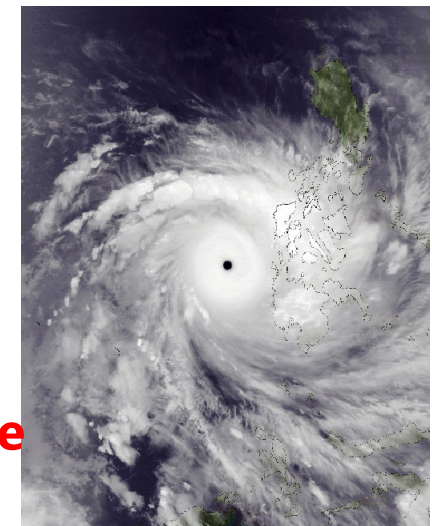


the way to measure body temperature

sparseness of stars

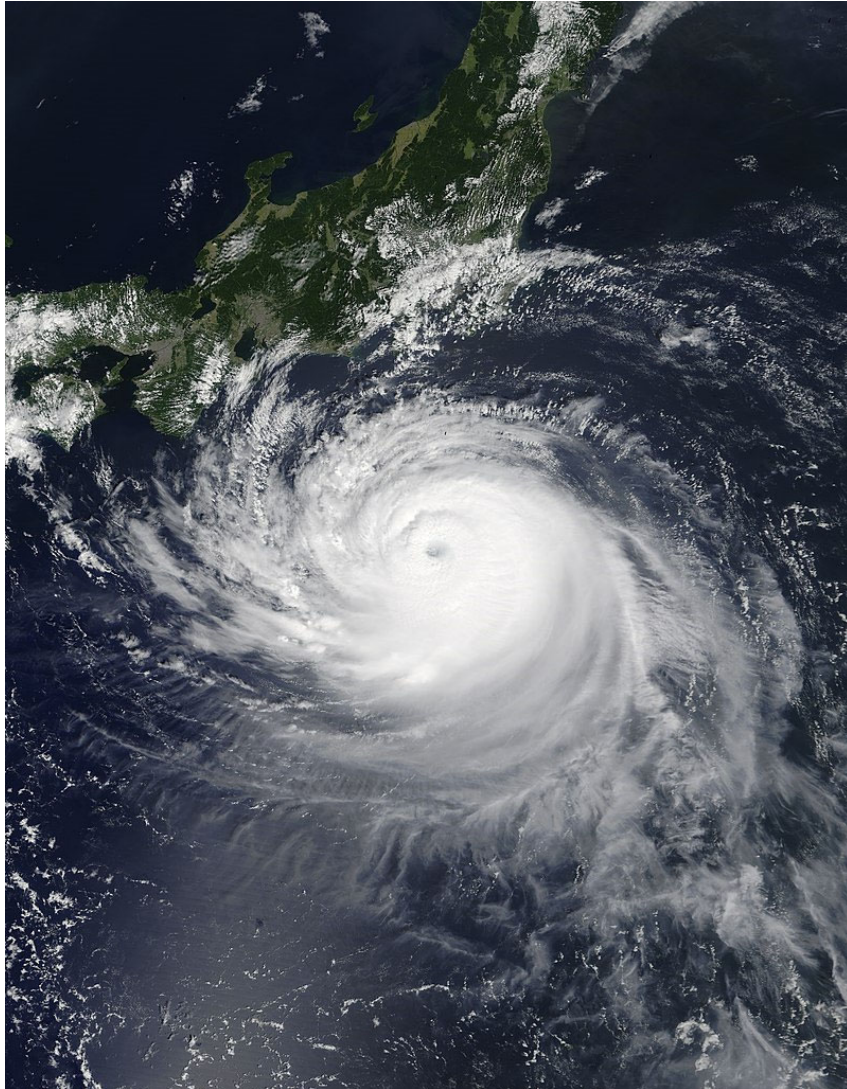


the way to measure TCs by satellites



今日の座学：
天気予報と
シミュレーションと
リモートセンシング

気象災害: ex. 令和元年房総半島台風



images from wikipedia

支配方程式（微分方程式）を解く

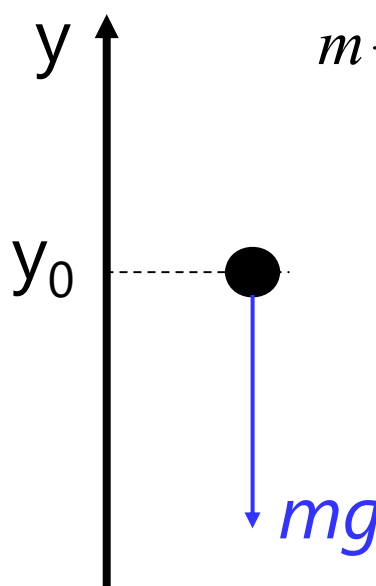
例) 運動方程式: 物体の運動を記述する方程式

$$m \frac{d^2 \mathbf{r}}{dt^2} = \mathbf{F} \xrightarrow{\text{積分}} \frac{d\mathbf{r}}{dt} \xrightarrow{\text{積分}} \mathbf{r}$$

運動方程式

\mathbf{F} : 力ベクトル \mathbf{r} : 物体の位置ベクトル

一番簡単な例: 自由落下運動

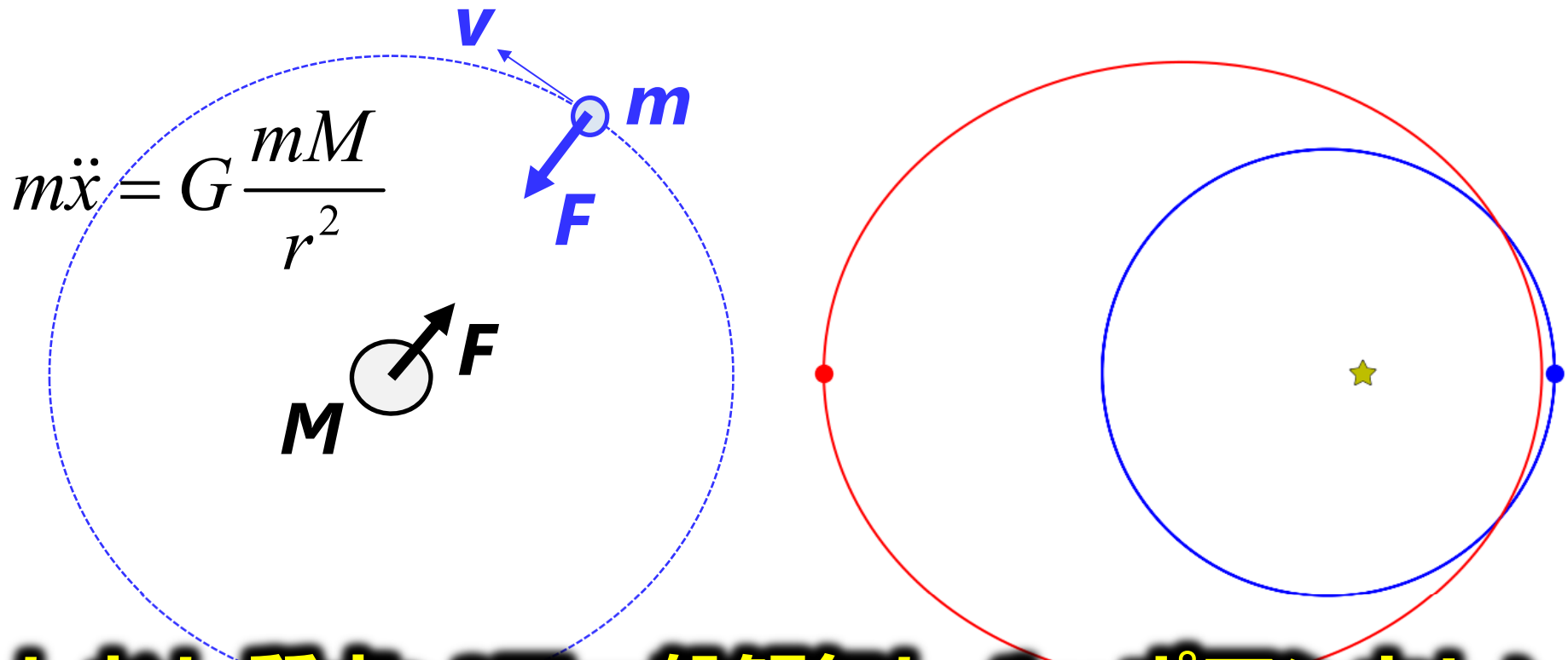

$$m \frac{d^2 y}{dt^2} = -mg \Leftrightarrow \frac{d^2 y}{dt^2} = -g \xrightarrow{\text{積分}} \frac{dy}{dt} = -gt + C = v$$
$$\xrightarrow{\text{積分}} y = -\frac{1}{2}gt^2 + Ct + C'$$

(i) 初期条件 $v(0) = 0$ より $C = 0$
(ii) 初期条件 $y(0) = y_0$ より $C' = y_0$ $\therefore y = -\frac{1}{2}gt^2 + y_0$

- ① 支配方程式を積分して初めて状態が分かる
- ② 予測には初期条件(積分定数)が必要

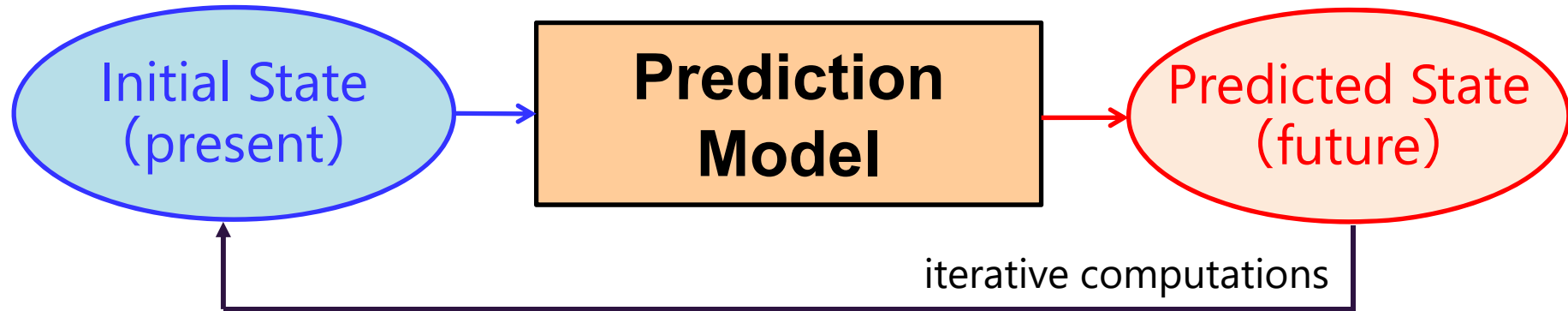
ニュートン力学とその限界 (求積解による予測)

世界は時間について2階の微分方程式 (運動方程式) で記述されていて積分で予測する (=求積する)

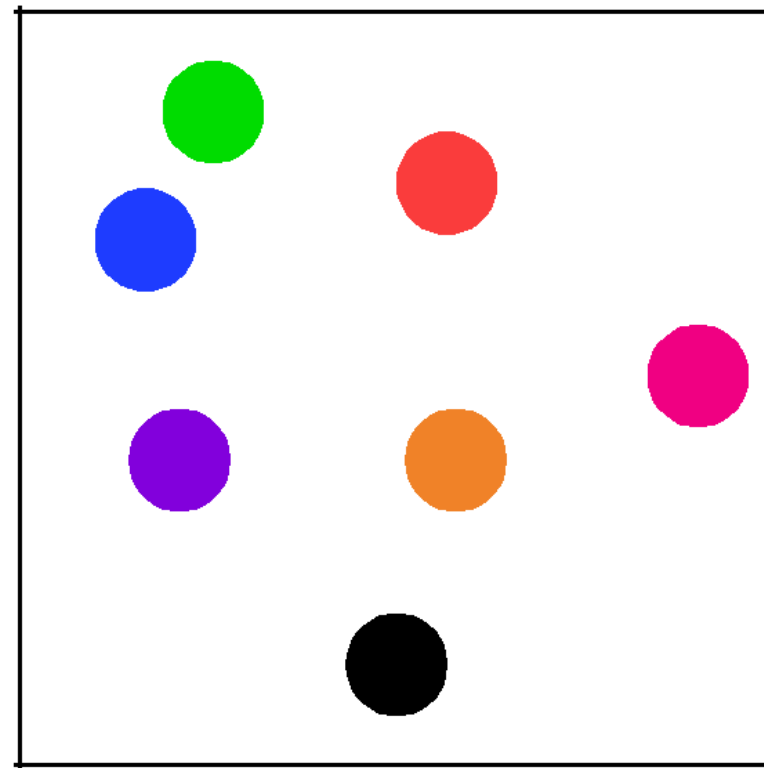


しかし質点 ≥ 3 で一般解無し (by ポアンカレ)
→ コンピュータを使った数値予測へ

Numerical Simulation w/ Computers



*an example
of billiards*



Numerical Weather Prediction

wind, temperature, humidity, pressure

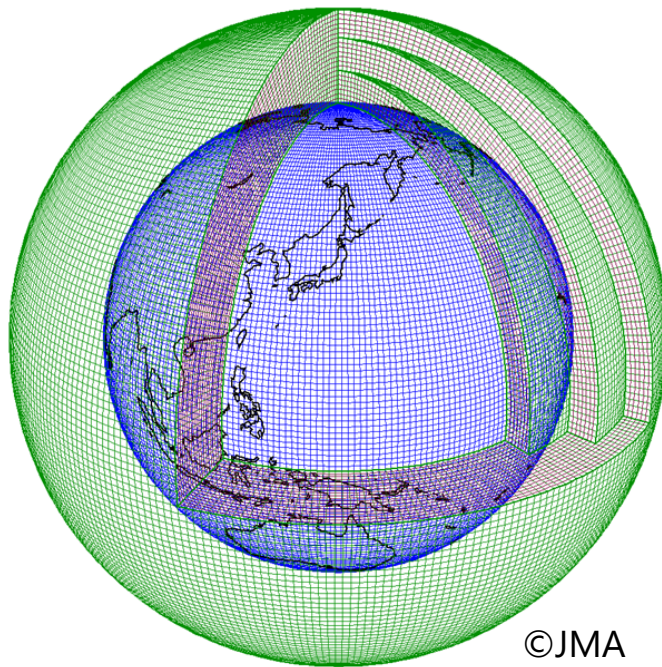


**Weather Prediction
Model**

wind, temperature, humidity, pressure

**Predicted State
(future)**

iterative computations



3-dimensional grids of atmosphere

• 水平方向の運動方程式

$$\frac{\partial u}{\partial t} - 2\Omega \sin \theta v + \frac{1}{a \cos \theta} \frac{\partial \phi}{\partial \lambda} = -\mathbf{V} \cdot \nabla u - \omega \frac{\partial u}{\partial p} + \frac{\tan \theta}{a} uv + F_u \quad (1)$$

$$\frac{\partial v}{\partial t} + 2\Omega \sin \theta u + \frac{1}{a} \frac{\partial \phi}{\partial \theta} = -\mathbf{V} \cdot \nabla v - \omega \frac{\partial v}{\partial p} - \frac{\tan \theta}{a} uv + F_v \quad (2)$$

• 熱力学第一法則の式

$$\frac{\partial c_p T}{\partial t} + \mathbf{V} \cdot \nabla c_p T + \omega \frac{\partial c_p T}{\partial p} = \omega \alpha + Q \quad (3)$$

• 質量保存則

$$\frac{1}{a \cos \theta} \frac{\partial u}{\partial \lambda} + \frac{1}{a \cos \theta} \frac{\partial v \cos \theta}{\partial \theta} + \frac{\partial \omega}{\partial p} = 0 \quad (4)$$

• 状態方程式

$$p\alpha = RT \quad (5)$$

• 静力学平衡の式

$$\frac{\partial \phi}{\partial p} = -\alpha \quad (6)$$

Simulated Global Precipitation



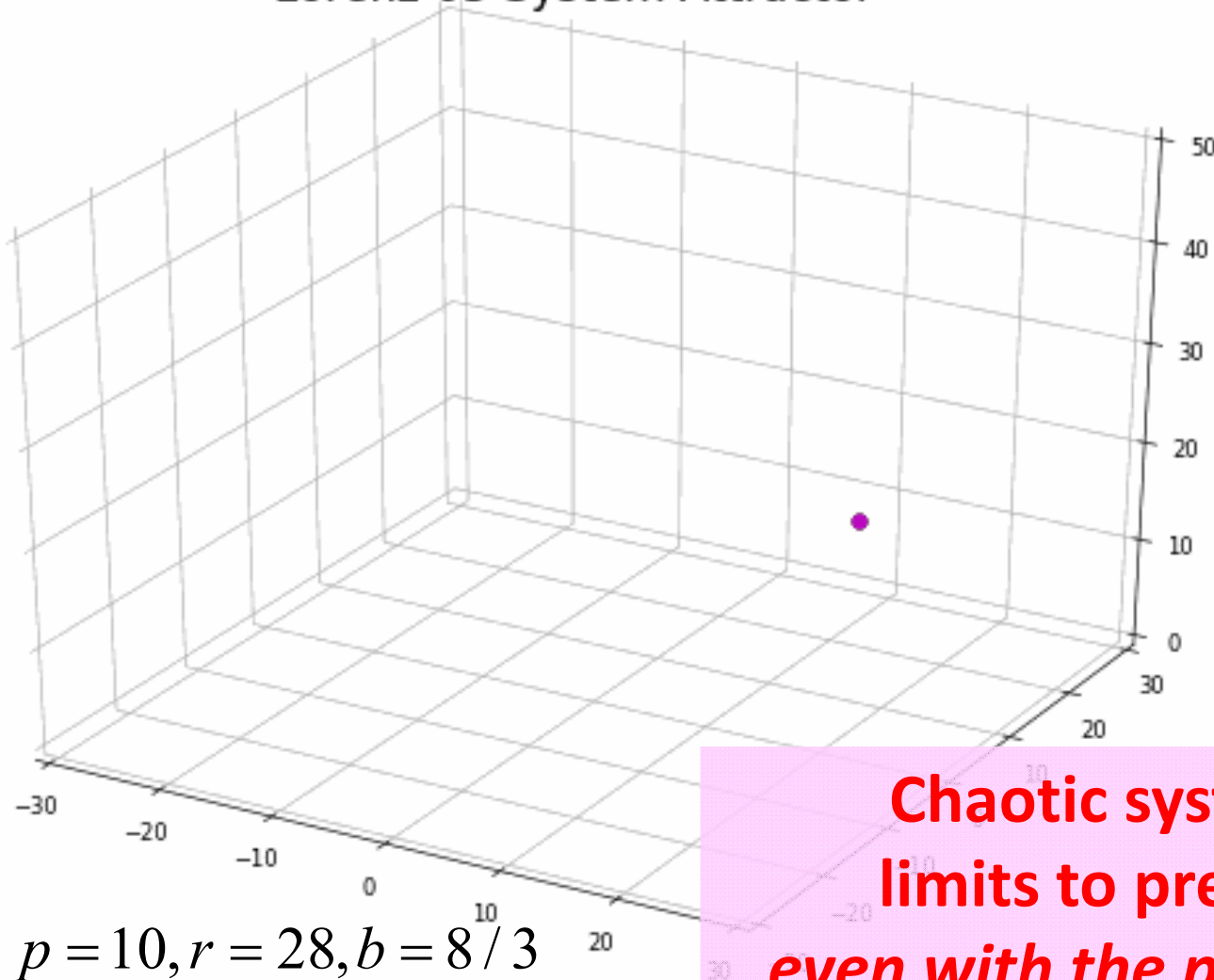
2014/05/25 00:00

Deterministic Chaos & Predictability

Edward Lorenz



Lorenz 63 System Attractor



Lorenz 63 model

$$\dot{x} = p(y - x)$$

$$\dot{y} = -xz + rx - y$$

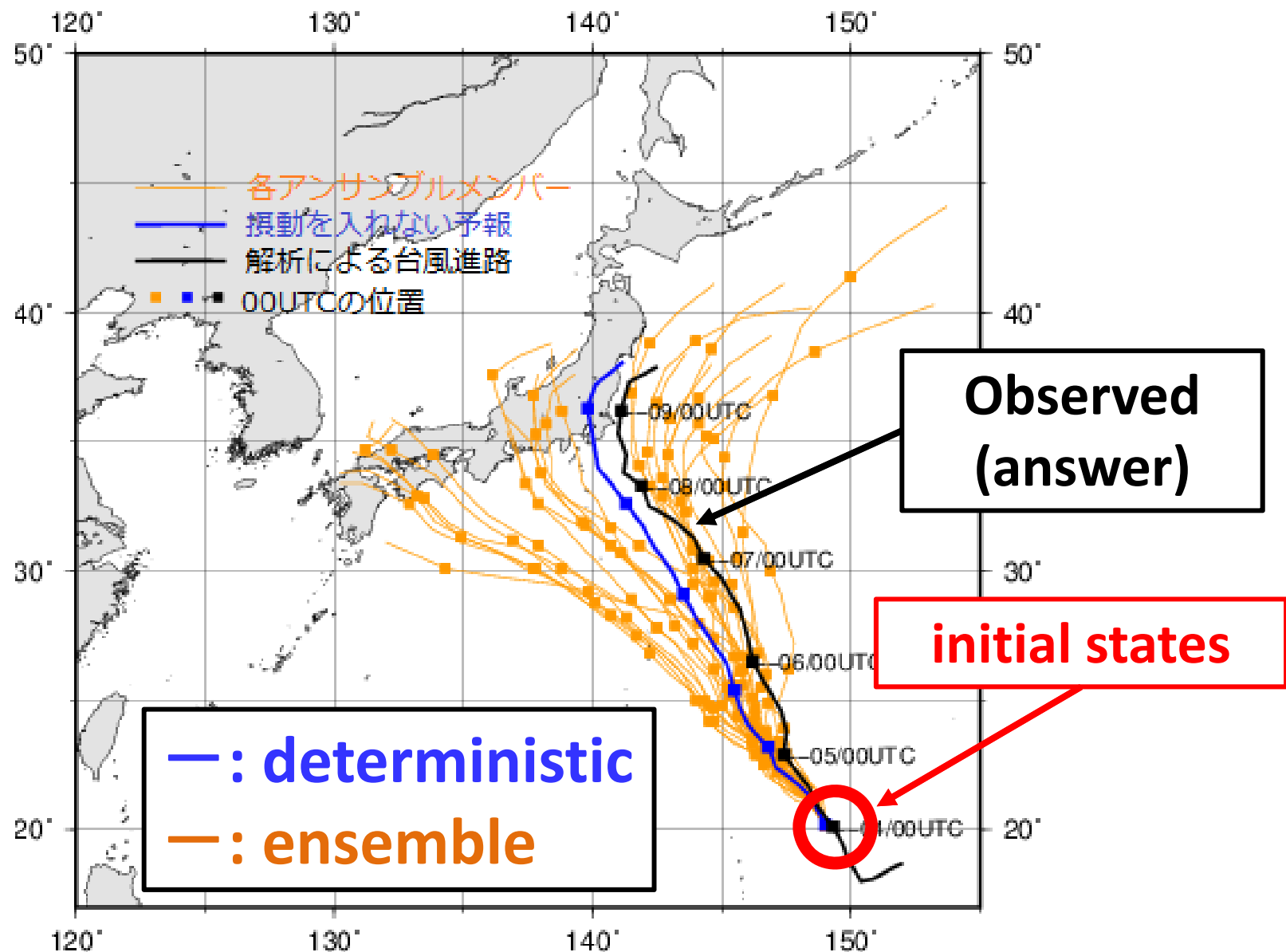
$$\dot{z} = xy - bz$$

$$p = 10, r = 28, b = 8/3$$

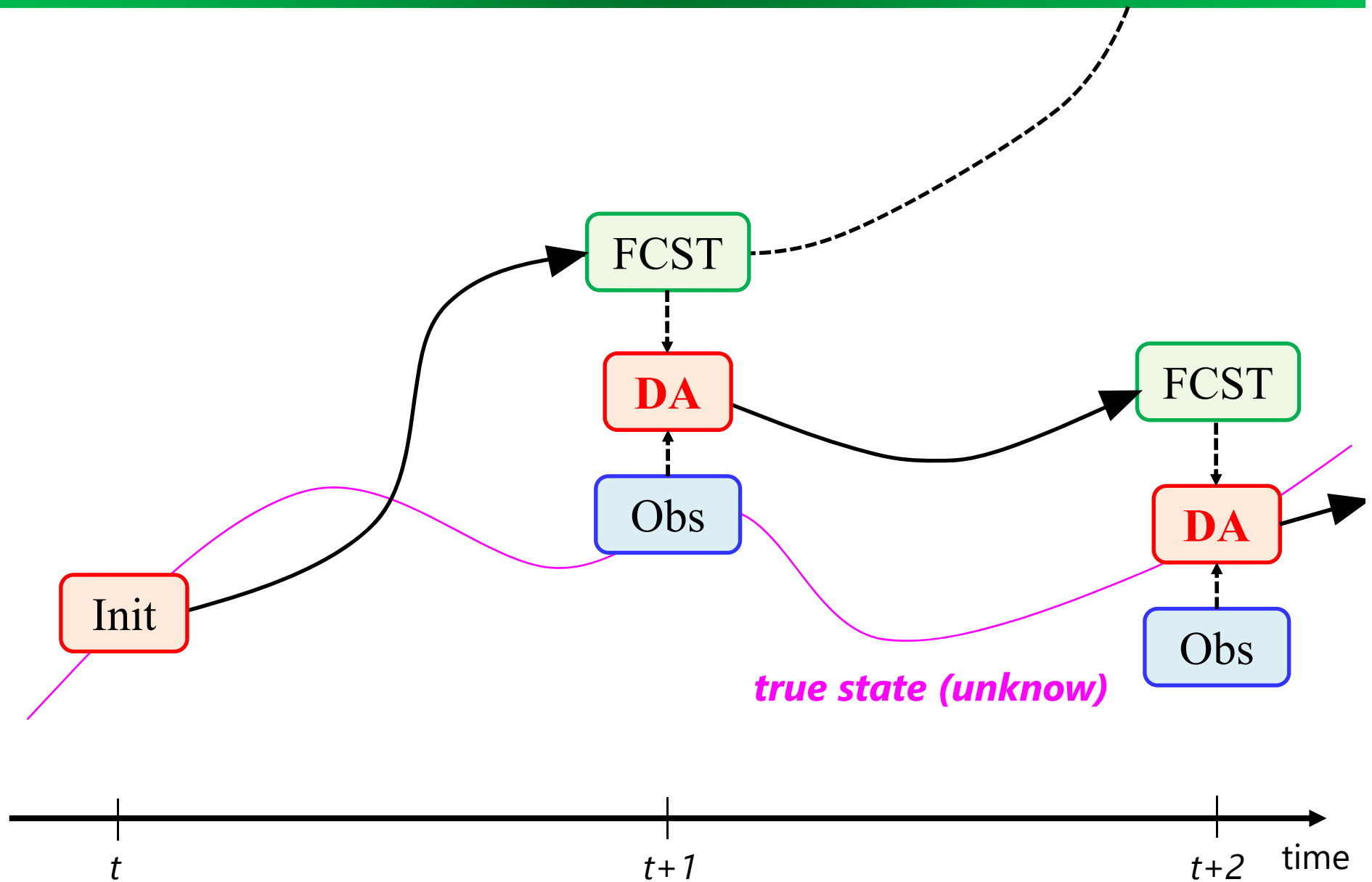
**Chaotic systems have
limits to predictability
even with the perfect model!!**

Initial Conditions :: $x=y=z=15.000, 15.001, 15.002, \dots, 15.009$

Ensemble Prediction: an example of typhoon



Numerical Weather Prediction



Global Observing System

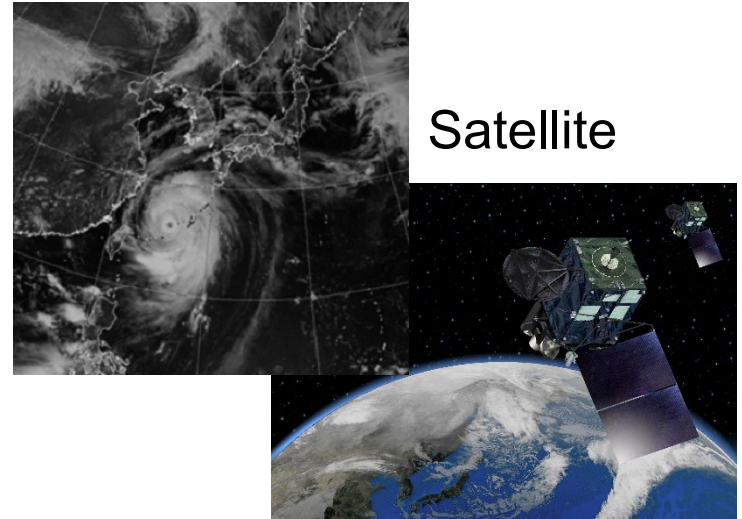
Radar



Aircraft



Satellite



Ship



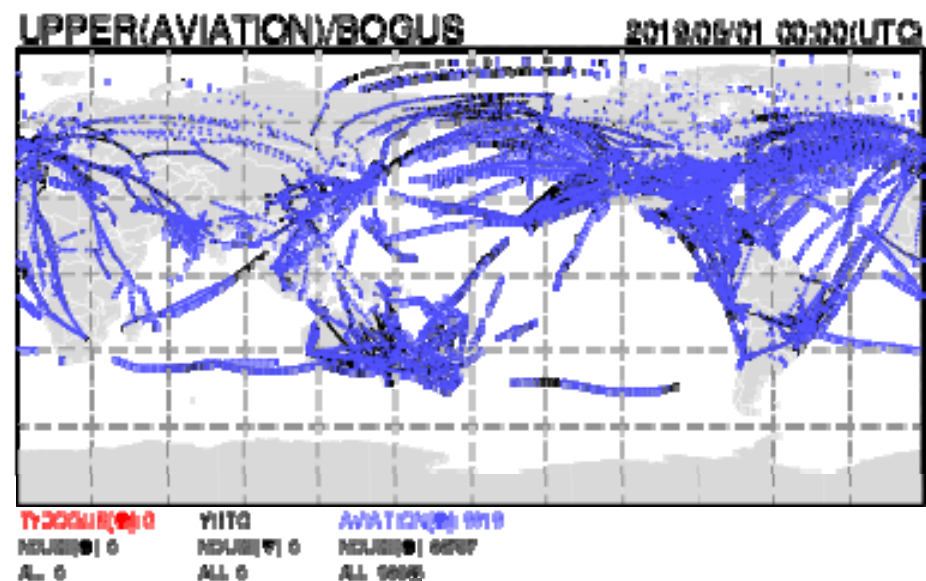
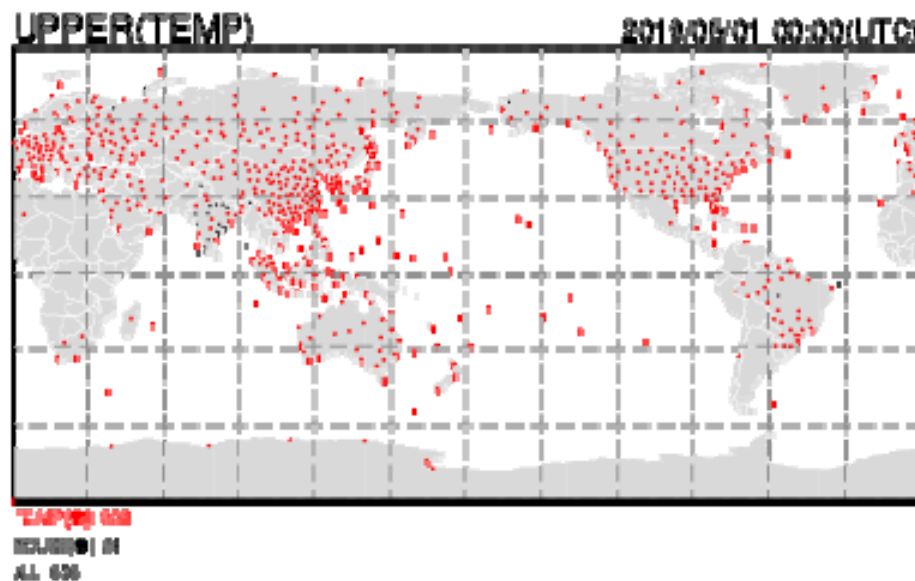
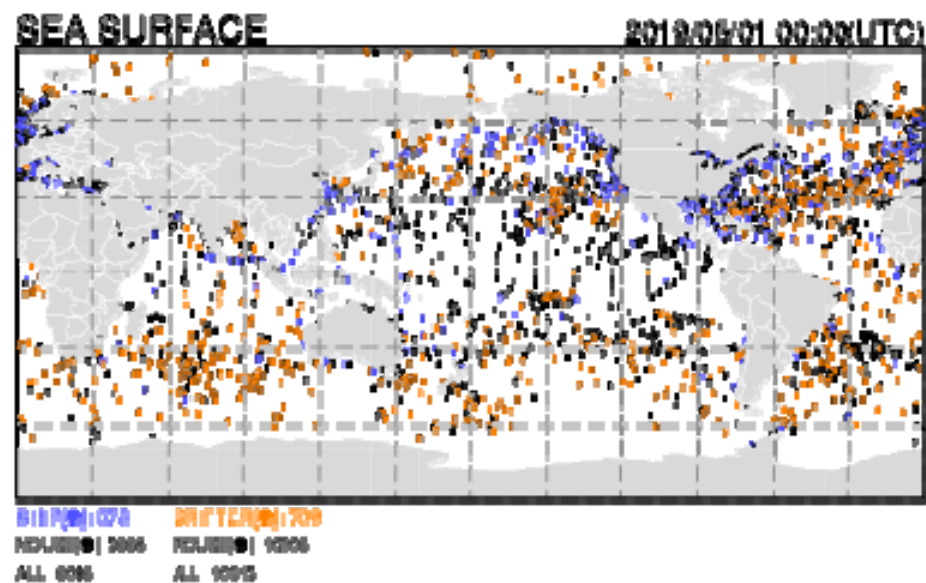
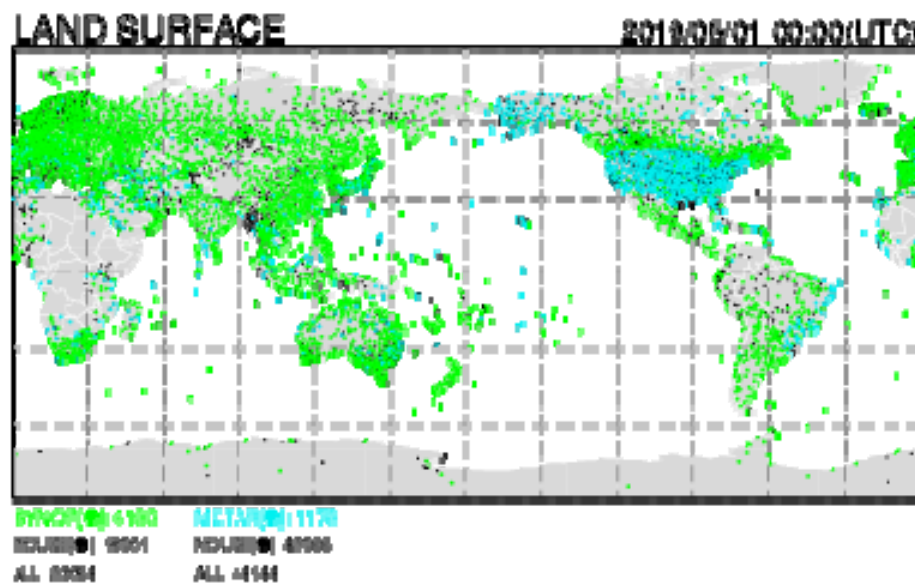
Buoy



Surface station

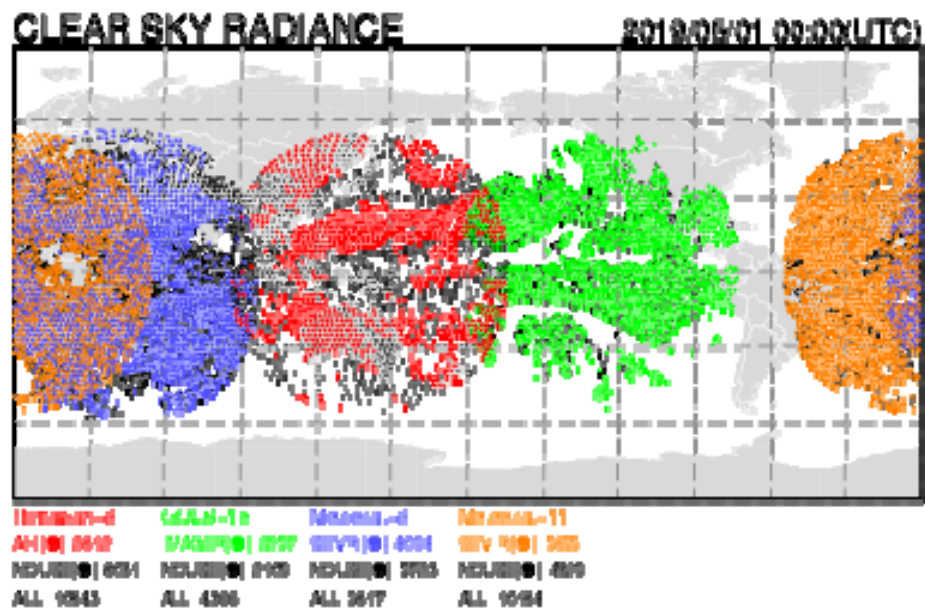
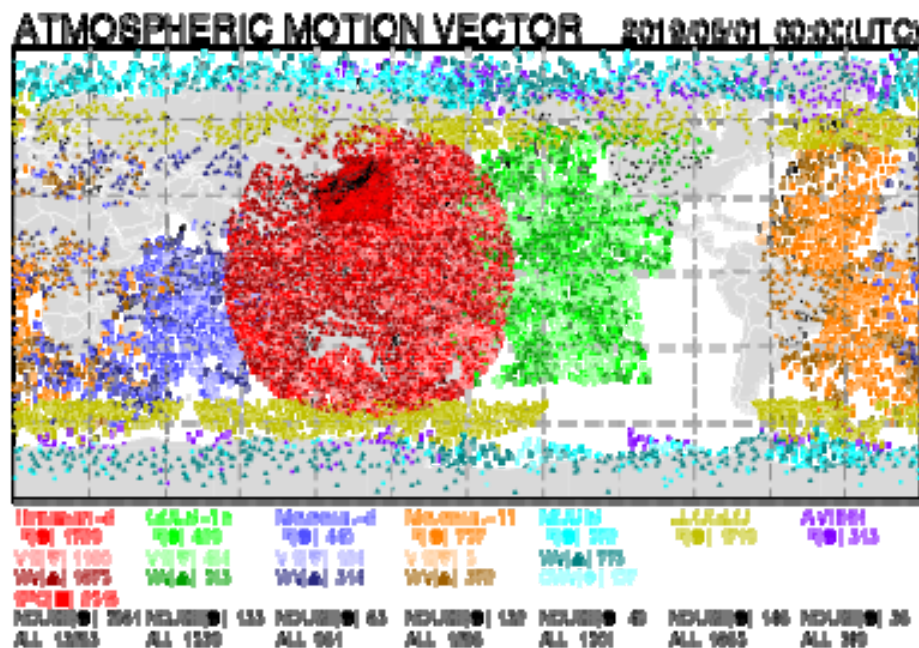
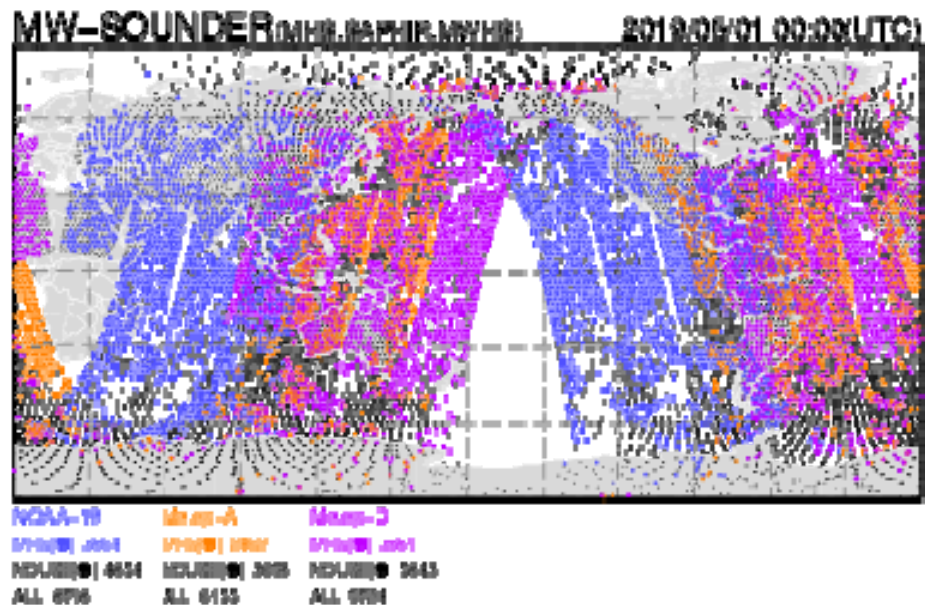
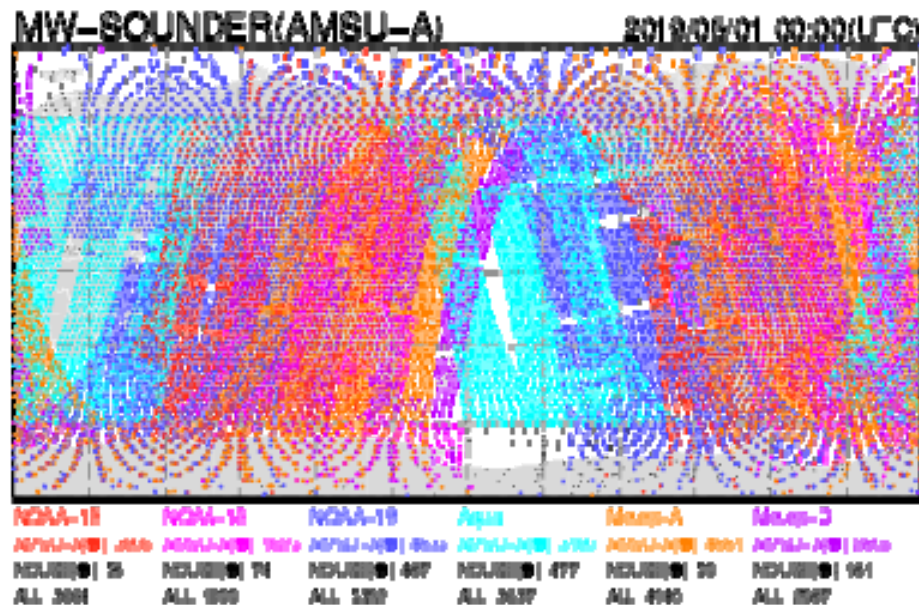


Observation Data in NWP: world's efforts!



courtesy of JMA (2019/05/01 00:00 UTC)

Observation Data in NWP: world's efforts!



courtesy of JMA (2019/05/01 00:00 UTC)

今回のまとめ

- **ガイダンス**

- 3人の教員が講義をします。
- 小槻担当回（50点分）はプログラミング演習と、それを踏まえたレポート提出です

- **雑談より**

- 大事ななのは、「結びつけ」る力。それをサポートできる様な講義を提供できる様努力します。

- **天気予報とリモートセンシング**

- シミュレーションとカオス
- 初期値を修正する、データ同化
- リモートセンシング人工衛星は必須の観測情報

初回Python演習

pythonプログラム演習の提出について

- **例えば（第1回目、numpyの課題例）**

乱数の入った5*5の配列を作成し、最大値最小値を取得した後に、平均0,標準偏差1に標準化した配列を作成せよ。

- **解答例**

```
1 # 5*5の乱数で初期化した配列を作成
2 Z = np.random.random((5,5))
3 print("最小値:", Z.min())
4 print("最大値:", Z.max())
5 # 標準化
6 Z = (Z - np.mean (Z)) / (np.std (Z))
7 Zmin, Zmax = Z.min(), Z.max()
8 print(Z)
```

- **こういった解答のソースコードをmoodleで各週、提出する（次ページ）。**

Google Colaboratory

- **Google Colaboratoryとは？**

- ブラウザ上で実行可能なpython環境
- ipython notebook形式
- GPUの利用が無料

- **基本的な使い方**

- 「Colaboratory へようこそ」で検索すると出てくるページが最も分かりやすい
- <https://colab.research.google.com/notebooks/intro.ipynb#>

課題に取り組む流れ

- **データの読込部分を含む導入ファイルをmoodleから取得**
 - 001_intro.ipynb, 002_intro.ipynb, 003_intro.ipynb ...etc
- **各自のgoogle driveにipynbファイルをアップロード**
 - ファイル管理のために、適当なフォルダを作成することを推奨
- **(初回のみ) google driveにcolaboratoryを紐づける**
 - ipynbファイルを右クリック→アプリで開く→アプリの追加
 - これで、G Suite Market placeというポップアップが開く
 - colaboratoryを探し、インストールをクリック
 - これで、「アプリで開く」でcolaboratoryで開けるようになる
- **ファイルをcolaboratoryで開き、課題に取り組む**
- **ファイル → ipynbをダウンロードでソースコードを取得**
- **ファイル名を変更し、課題アップロード**
 - 拡張子前に学籍番号を記入して、ファイル名を編集
 - 例えば、 001_intro_18A9999B.ipynb など

Google Colaboratory
実演中

初回は以上です。
プログラミング演習に取り組み、
ソースコードを提出してください。

次回までに、教科書の
1章：太陽系の中の地球
2章：大気の鉛直構造
を読んできて下さい

